Problem Set 1: GWAS

The goal of this problem set is to familiarize you with running a basic genome-wide association study (GWAS) on a sample dataset, and analyzing the results. You will be provided with a genotyping dataset from a previously published Parkinson's GWAS.

First, read the publication by Hon-Chung Fung, et al (https://www-sciencedirect-com.laneproxy.stanford.edu/science/article/pii/S1474442206705786?via%3Dihub), "Genome-wide Genotyping in Parkinson's Disease and Neurologically Normal Controls: First Stage Analysis and Public Release of Data" (you'll most probably need to log in with your Stanford SUNet to access the linked page).

This publication describes a genome-wide association study on Parkinson's Disease. More than 408,000 single nucleotide polymorphisms (SNPs) were measured (or genotyped) across patients with Parkinson's Disease and normal control individuals. Each SNP is a potentially differing nucleotide between individuals. Recall that there are estimated to be as many as 10 million SNPs in the human genome, so this collection does not encompass all of them.

For this assignment, you will need to write code in R or Python to answer the questions that follow. Please fill out the following markdown file (if you're using R) or notebook (if you're using Python) with appropriate code or written answers. For those of you unfamiliar with notebooks, notebooks allow for reproducible research, a better "sandbox" to play with variables, and increased readability of code. You can make "cells" of text ("markdown cells" or just text for markdown files) interspersed with your code ("code cells") in order to explain what is happening in your code.

1) Jupyter users: In order to edit any cell, double click on it within the Jupyter environment. Press (Shift + Enter) to "run" the cell, or compile it such that it turns back into text, or runs the code in the cell, depending on what type of cell you are in. Most notebooks automatically move to the next cell after running the previous one. Results should pop up right below the code. Submit the full notebook, with code and written answers.

2) RStudio users: Edit text as necessary as you would a Word document. Fill in code cells and press (Ctrl + Enter) to "run" the code cell (on Mac, (Command + Enter) also works). Results should pop up right below the code. When done, "Knit" the results to a

PDF (there should be a button at the top of your screen called "Knit" with a dropdown menu) and submit, with code and written answers.

Please submit your code along with your answers to the questions.

Part 1: The Original Study

Read through the paper provided to find the answers to the following questions.

1.1 (2 pts)

How many cases and controls were included? Does this seem like enough samples to identify rare variants? Explain why or why not.

They used 267 cases and 270 controls. This is not enough to identify rare variants, but is enough to identify potential common causal variants. Rare variants only appear in a small percentage of a population, so to maximize the chance of finding rare variants, they would want to have as many cases as possible.

1.2 (2 pts)

What genotyping chip was used? How many variants are included on the chip?

They used Illumina Infinium I and HumanHap300 assays. According to the paper, they used more than 408,000 unique SNPs on the chip. Specifically, 408,803 SNPs coming from 109,365 gene-centric SNPs (Infinium I) and 317,511 haplotype tagging SNPs(HumanHap300) with an overlap of 18,073 SNPs.

1.3 (2 pts)

How many samples were not included due to quality control filtering? Why weren’t they included? (leniency)

They excluded 7 samples due to quality filtering. They weren't included because they consistently had a call rate below 95% from repeated DNA aliquots. Two of those seven samples had also become contaminated.

Part 2: Running a genome-wide association study - PLINK

PLINK (http://zzz.bwh.harvard.edu/plink/) is a commonly used software tool for running various types of genomic studies, and is one of the most commonly used tools for running GWAS. You have been provided with a few files: `parkinsons.map`, `parkinsons.ped`, `1kg_chr1.bed`, and `multiallelic.missnp`. These are standard file formats used as inputs to PLINK. You have also been provided with some other files (`common_snps.txt`, `igsr_samples.tsv`) that will be useful later in this problem.

Note: PLINK 2 was recently made available but is not yet as well documented as PLINK 1. **For this analysis, please use PLINK 1.9**. PLINK 2 contains additional features that can be used in more advanced analyses and runs much faster than PLINK 1 on large datasets, but is not necessary for this analysis.

Running PLINK

Install the stable build of PLINK 1.9 (https://www.cog-genomics.org/plink2) on your computer and run following command in the directory in which you have stored the files provided:

```
plink --allow-no-sex --file parkinsons --no-fid --no-sex --no-parents --assoc --out parkinsons
```

Several files will be outputted, including `parkinsons.assoc`, which will be used in the next section.

2.1 (4 pts)

For each file (`.map`, `.ped`, `.bed`, `.missnp`), find the description of the filetype on the PLINK website and describe its contents.

.map: Variant information file. A text file with no header, and one line per varaint with 3-4 fields.
1) Chromosome code
2) Variant identifier
3) Position in morgans or centrimorgans
4) Base Pair coordinate

.ped: Sample pedigree information and genotupe cells. No header. One line per sample with 2V+6 fields where V is the number of variants.
First 8 columns:
1) Family ID ('FID')
2) Within-family ID ('IID'; cannot be '0')
3) Within-family ID of father ('0' if father isn't in dataset)
4) Within-family ID of mother ('0' if mother isn't in dataset)
5) Sex code ('1' = male, '2' = female, '0' = unknown)
6) Phenotype value ('1' = control, '2' = case, '-9'/'0'/non-numeric = missing data if case/control)
7 & 8) Allele calls for the first variant in the .map file

.bed: Binary biallelic genotype table. Primary representation of genotype calls at biallelic variants. The first three bytes should be 0x6c, 0x1b, and 0x01 in that order. The rest of the file is a sequence of V blocks of N/4 (rounded up) bytes each, where V is the number of variants and N is the number of samples.

.missnp: A list of variants with more than two alleles referred to by their rs accession numbers.

2.2 (2 pts)

These files were prepared as input for PLINK. The file containing variant data is `parkinsons.ped`. Compare and contrast ped format to variant call format (https://samtools.github.io/hts-specs/VCFv4.2.pdf) (or VCF), another commonly used format for storing variant data.

VCF is the format for the 1000 Genomes Project while .ped was made more for plink. VCF has a lot of the same information as .ped and the format can even be converted to a .ped file using plink. VCF files also include more metadata in the form of 5+C header lines where C is the number of chromosomes.

2.3 (2 pts)

We will be using the default association test provided by PLINK. What statistical test is run by default, and why is this an appropriate option for the data we have?

The defulat statistical test that is run is a 1 df chi-square allelic test. This is appropriate because we are able to see the prevalence of SNPs in different affected individuals and check to see the statistical significance of the findings in multiple samples and gather a p-value.

2.4 (2 pts)

What are the other options for running association tests using plink?

- Stratified case/control analyses
- Quantitative phenotypes
- Regressions with multiple covariates
- LASSO Regression
- Linear mixed model association
- Nonrandom Missingness Test

2.5 (2 pts)

One commonly used statistical test for binary trait GWAS is logistic regression. What advantages does logistic regression have over the default option?

Logistic regression gives better controls for a test and better confidence in the test. Logistic regression also offers a way to model a variable as an out come of one or more variables with some probability.

Part 3: Population stratification check

Population stratification is when genetic population structure is correlated with outcome. A silly example: let's say that our phenotype of interest is eating Swiss cheese. A whole bunch of genes that are associated with being a Swiss person, then, would show up as hits in a GWAS for this silly trait. We want to avoid this kind of spurious correlation in our studies.

Population stratification is a problem associated with false positives that is encountered routinely in GWAS, especially when dealing with large datasets.

Let's make sure that population structure will not be a problem with our dataset. It has been shown that principal component analysis (PCA) is a helpful tool to determine ancestry and thus correct for population stratification in large GWAS (https://www.nature.com/articles/ng1847). It is a method that attempts to represent the complete samples-by-sites matrix in a low-rank setting; that is, it attempts to reduce the dimensionality of the original matrix by capturing the major axes of variation through finding the eigenvalues and corresponding eigenvectors of the covariance matrix. It turns out that the first few principal components of large GWAS align nicely with longitude and latitude in Europe, and are good measures of ancestry globally (https://www.nature.com/articles/nature07331).

The larger the eigenvalue, the more the eigenvector (or principal component) accounts for variability in the original dataset. PCA extracts the first kk eigenvalues and eigenvectors, with kk being 20 in the case of `plink`.

If this material is not review to you, it is highly encouraged that you read the specifics of how PCA works here (http://www.math.union.edu/~jaureguj/PCA.pdf); it shows up over and again in translational bioinformatics.

Luckily for us, PLINK comes with a built-in PCA option. In order to see whether or not our dataset is uniform in ancestry, we will combine our dataset with the 1000 Genomes dataset and run PCA on the joint dataset. For those of you that are unfamiliar, the original paper by the 1000 Genomes (1kG) Consortium can be found here (https://www.nature.com/articles/nature15393). It was a landmark publication that "set out to provide a comprehensive description of common human genetic variation by applying whole-genome sequencing to a diverse set of individuals (2504) from multiple (26) populations" (and 5 superpopulations).

We want to use a good number of sites (on the order of 10s of thousands) common to both datasets (the Parkinsons dataset was taken on a specific array that contains a small number of sites), and want to merge the individuals that are in 1KG with these individuals across these common sites.

For computational tractability, we have only decided to use Chromosome 1 of the 1kG dataset, which provides a sufficient number of common sites across both datasets.

3.1

We have generated a list of SNPs common to both of the datasets that must be **extracted** (`common_snps.txt`) as well as a list of multiallelic sites that must be **excluded** (`multiallelic.missnp`). Those of you interested in how these were generated can find the details here (https://docs.google.com/document/d/15wmCpKaFTlNFt4tjwGemfEhuJtAPzIKnufQS-LiScXI/edit?usp=sharing).

Write two `plink` commands to do both the extraction and the exclusion on 1) `parkinsons.ped` and 2) `1kg_chr1.bed`, and save the results to a new file. Relevant `plink` commands you might find useful: `--file`, `--bfile`, `--extract`, `--exclude`, `--make-bed`, `--out`.

```
plink --allow-no-sex --file parkinsons --no-fid --no-sex --no-parents --extract common_snps.txt --exclude multiallelic.missnp --make-bed --out parkinsons_relevant_snps
plink --allow-no-sex --bfile 1kg_chr1 --no-fid --no-sex --no-parents --extract common_snps.txt --exclude multiallelic.missnp --make-bed --out 1kg_relevant_snps
```

3.2

Finally, merge the resultant files. Relevant commands: `--bfile`, `--bmerge`, `--make-bed`, `--out`.

```
plink --bfile parkinsons_relevant_snps --bmerge 1kg_relevant_snps --make-bed --out merged_snps
```

3.3 (3 pts)

Finally, run PCA on the merged file. You should get two important files as results - `plink.eigenval` and `plink.eigenvec`.

Load `plink.eigenvec` and `plink.eigenval` into pandas and take a look. What do these files contain?

```
plink --allow-no-sex --bfile merged_snps --pca
```

In [1]:

```
import pandas as pd
eigenval = pd.read_csv('plink.eigenval', header=None)
display(eigenval.head())

names = ['Sample name', 'sn2', 'PC1', 'PC2', 'PC3', 'PC4', 'PC5', 'PC6', 'PC7', 'PC8', 'PC9', 'PC10', 'PC11', 'PC12',\
        'PC13', 'PC14', 'PC15', 'PC16', 'PC17', 'PC18', 'PC19', 'PC20']
eigenvec = pd.read_csv('plink.eigenvec', header=None, delim_whitespace=True, names=names)
display(eigenvec.head())
```

	0
0	271.1520
1	153.2180
2	35.4148
3	26.7031
4	12.5945

	Sample name	sn2	PC1	PC2	PC3	PC4
0	HG00096	HG00096	-0.010963	-0.016852	-0.003983	-0.001950
1	HG00097	HG00097	-0.009644	-0.018543	-0.004391	-0.004544
2	HG00099	HG00099	-0.010698	-0.018117	-0.008796	-0.002710
3	HG00100	HG00100	-0.011708	-0.017607	-0.004536	0.003537
4	HG00101	HG00101	-0.011003	-0.018810	-0.007955	0.003605

5 rows × 22 columns

The plink.eigenvalue file contains information on the 20 principal components and how much of the variation in th data is explained by each principal component. The plink.eigenvector file contains individuals (coded in the first two columns) and the loadings/weights they have on each of the 20 principal components.

3.4 (15 pts)

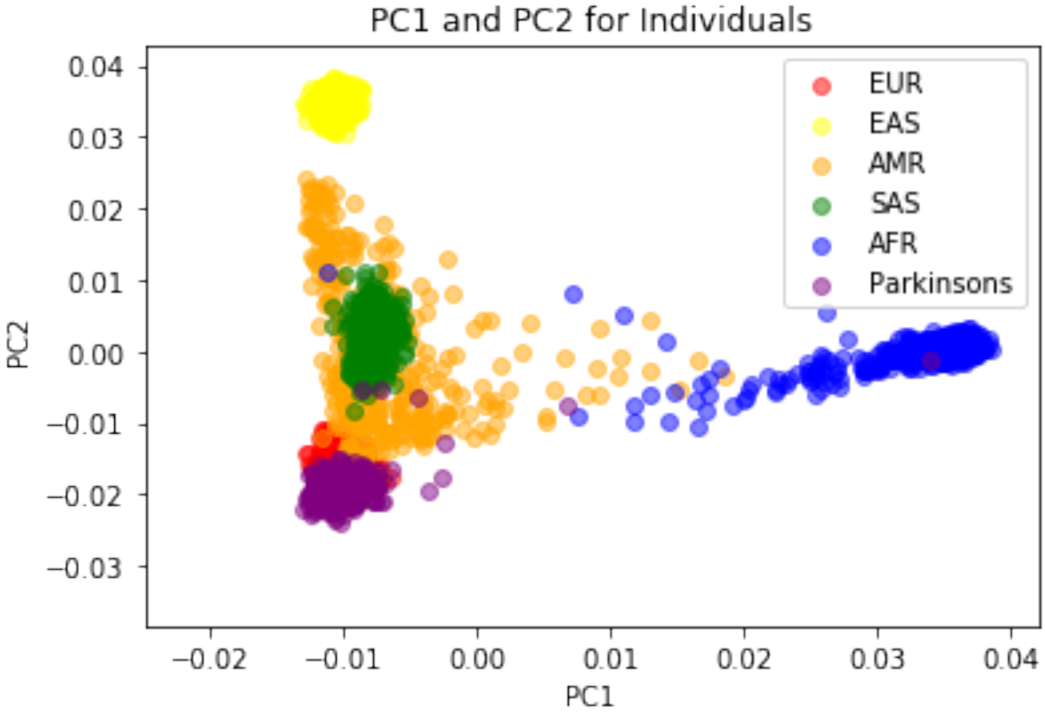
- Left-join the `plink.eigenvec` table you loaded into pandas earlier with `igsr_samples.tsv` in order to get the superpopulation code per individual.
- All of the individuals that do not have a superpopulation code in the resultant table are from the Parkinson's dataset. Label them as such by filling in the NA values in this column with the string `Parkinsons`.
- Do a scatterplot with PC1 on the x-axis and PC2 on the y-axis for all individuals, coloring the dots by superpopulation code. Visualizing these two PCs allows us to view the two largest axes of "spread" in this dataset. While `matplotlib` does not easily handle a color column, the `seaborn` package has a function `lmplot` that does handle this well. Useful input parameters you might want to enable/disable in this function: `hue`, `fit_reg`.

In [2]:

```
#YOUR CODE FOR 3.4 HERE
import matplotlib.pyplot as plt
igsr_samples = pd.read_csv('igsr_samples.tsv', sep='\t')
result = pd.merge(eigenvec, igsr_samples[['Sample name', 'Superp
opulation code']], on='Sample name', how='left')
result.fillna(value='Parkinsons', inplace=True)

colors = pd.factorize(result['Superpopulation code'])
result['Colors'] = colors[0]

codes = result.Colors.unique()
color_list = ['red', 'yellow', 'orange', 'green', 'blue', 'purpl
e']
for code in codes:
    sub_frame = result[result['Colors'] == code]
    plt.scatter(sub_frame['PC1'], sub_frame['PC2'], c=color_list
[code],
                label=sub_frame['Superpopulation code'][sub_fram
e.index[0]], alpha=0.5)
plt.title('PC1 and PC2 for Individuals')
plt.xlabel('PC1')
plt.ylabel('PC2')
plt.legend()
plt.show()
```



3.5 (2 pts)

From this plot, which population does the Parkinson's dataset seem closest to? Does this make sense given the description of the people in this study?

The Parkinson's dataset is closest to the European cluster from the 1K Genome project. This makes sense because the study said they used white men between the ages of 55 and 84.

3.6 (2 pts)

There are quite a few people in the 1kG dataset that are of admixed ancestry (not purely of one superpopulation, but in the space between them). What problems could arise when doing studies in admixed populations?

With admixed studies you get varied ancestral/racial genomes that become harder to control for. It is easier to make conclusions on highly controlled data.

Part 4: Analysis

Use `parkinsons.assoc` from Part 2. Use R or Python to write code to answer each of the questions.

4.1 (2 pts)

List and describe each column included in the assoc file.

In [3]:

```
assoc = pd.read_csv('parkinsons.assoc', delim_whitespace=True)
assoc.head()
```

Out[3]:

	CHR	SNP	BP	A1	F_A	F_U	A2	CHISQ	
0	1	rs2977670	763754	C	0.029880	0.01245	G	3.5780	0
1	1	rs7526509	1038797	T	0.011110	0.00369	A	2.0300	0
2	1	rs3934834	1045729	T	0.133100	0.14770	C	0.4684	0
3	1	rs3766193	1057093	G	0.431500	0.42250	C	0.0890	0
4	1	rs12096277	1057521	G	0.001852	0.00369	A	0.3306	0

CHR: The chromosome the variant is on.
SNP: Variant Identifier
BP: BAse-pair coordinate
A1: Allele 1
F_A: Allele 1 frequency among cases
F_U: Allele 1 frequency among controls
A2: Allele 2
CHISQ: Allelic test chi-square statistic
P: Allelic test p-value
OR: odds(allele 1 | case) / odds(allele 1 | control)

4.2 (5 pts)

How many rows have NAs for the pp-value? Why might this be?

In [4]:

```
num_nan = len(assoc) - assoc['P'].count()
print(num_nan)
```

```
905
```

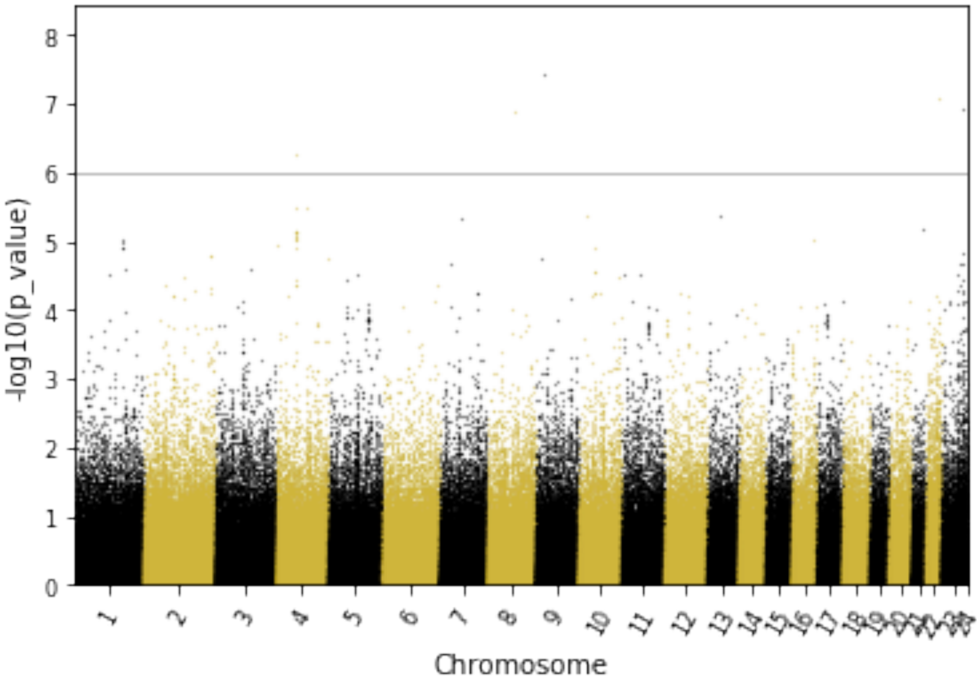
This data could be NA becuase the information was lost or corrupted in some way. The phenotype also could be missing for those individuals and so it did not have the information to calculate the pvalue.

4.3 (10 pts)

Create a manhattan plot to visualize the results of the GWAS. Describe each of the axes in your own words. If you are using Python, use the Pyhattan (https://github.com/Pudkip/Pyhattan) package.

In [5]:

```
from Pyhattan import FormatData, GenerateManhattan
pyhattan_obj = FormatData('parkinsons.assoc', sep='\s+', chromosome='CHR', p_value='P')
GenerateManhattan(pyhattan_obj, colors=['black', '#CFB53B'])
```



The x-axis represents chromosome postions/locations in the genome. The y-axis shows how strongly associated the SNP is with the phenotype. The smaller the p value (larger the -log of the p value), the more strongly associated.

4.4 (2 pts)

For GWAS, a Bonferroni correction is typically used to account for multiple hypothesis testing. Describe the method and why this is necessary.

The Bonferroni correction is a method of checking the statistical relevance of a finding when several dependent/independent statistical tests are being run at the same time. It sets the cutoff for significant pvalues to be proportional to the number of tests. Specifically setting the cutoff to be $\alpha/n\alpha/n$. This is necessary to prevent a high number of false positives that come from an increasingly high number of tests.

4.5 (5 pts)

The widely accepted Bonferroni adjusted pp-value threshold for GWAS is of $5x10^{-8}$ $\mathbf{5x10^{-8}}$. Using this threshold, how many significant hits are there? For the three variants with the smallest pp-values (regardless of significance), search the internet for functional information and list your findings (e.g. what genes they are related to, whether they are coding or non-coding, etc). If they are associated with a gene, describe the function of the gene and whether it is logical that it could be related to Parkinson's.

In [6]:

```
threshold = 5*(10**-8)
relevant_snps = assoc[assoc['P'] < threshold]
print(len(relevant_snps), 'SNP(s) with p-values less than thresh
old')
```

```
1 SNP(s) with p-values less than threshold
```

In [7]:

```
my_thresh = 5*(10**-7)
snps = assoc[assoc['P'] < my_thresh]
snps.head()
```

Out[7]:

	CHR	SNP	BP	A1	F_A	F_U	A2	CHI
201737	8	rs7846412	92113704	A	0.19820	0.3714	G	27.8
215059	9	rs10963676	18612043	G	0.05702	0.1704	T	30.3
395278	22	rs5766565	43943221	T	0.25560	0.4096	G	28.7
405852	23	rs5958478	123244763	A	0.30710	0.5130	C	28.0

rs5766565: not much is known about this variant and it is not clearly associated with Parkinson's.

- found in non-coding intron
- located on chr22
- Gene: KIAA0930
    - KIAA0930 (KIAA0930) is a Protein Coding gene
- Genic Upstream Transcript Variant
- Not reported in clinvar

rs10963676: the ADAMTSL1 gene is a part of a familt of protease enzymes that have been demonstrated to play an important role in Arthritis along with other disorders. This leads me to believe that there may be some connection with Parkinson's.

- found in non-coding intron
- located on chr9
- Gene: ADAMTSL1
  - This gene encodes a secreted protein and member of the ADAMTS
  - Diseases associated with ADAMTSL1 include Microcephaly-Facial Dysmorphism-Ocular Anomalies-Multiple Congenital Anomalies Syndrome and Quebec Platelet Disorder
- Genic Upstream Transcript Variant
- Not reported in clinvar

rs5958478: given that the gene(TENM1) is associated with connectivity within the nervous system, it is logical to think it may be related to Parkinson's.

- found in non-coding intron
- located on X chromosome
- Gene: TENM1
  - The protein encoded by this gene belongs to the tenascin family and teneurin subfamily. It is expressed in the neurons and may function as a cellular signal transducer.
  - Involved in neural development, regulating the establishment of proper connectivity within the nervous system.
  - Diseases associated with TENM1 include Anosmia, Isolated Congenital and Anal Margin Carcinoma
- Genic Downstream Transcript Variant
- Not reported in clinvar

4.6 (2 pts)

How do these results compare to the original study? Why might they be different? Note: we expect your results to be quite different. Don't panic!

It seems that none of the variants my analysis flagged as potentially significant appeared in the study. The study also seems to have had most of its candidate SNPs rejected after the Bonferonni correction even after getting a slightly less stringent cutoff than what we used. We may have found some SNPs with more significant p-values due to us not being limited by the SNPs that the chips they are using will catch. The difference could also be accounted for by different methodologies/calculations.